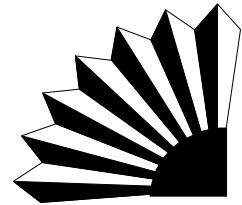


# the Technical Broadcast



Published by the Department of Information Services

Summer 1995

## Legent's ADASTRIP ADABAS Backup Tape Extract Utility

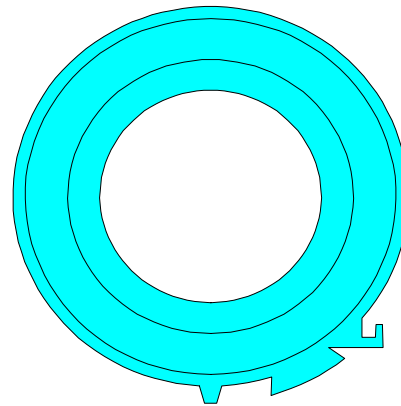


by Mike Hauser

**T**he ADASTRIP utility extracts data from the ADASAV backup and can build over 100 sequential output files. Each output file is built from a specified ADABAS file on the backup dataset and contains decompressed fields from the ADABAS files.

Comprehensive selection criteria enable the output file to contain almost any desired subset of dumped records. The extracted data can be the entire decompressed record suitable for reloading using the ADACMP utility. Any subset of the fields in each record can be nominated for extraction. It can do length overrides and field occurrence limits. Field occurrence limits allow easier processing of PE and MU fields in the extract using NATURAL. The extracted data is a sequential dataset with a record format of FB or VB.

ADASTRIP has been used to extract large amounts of data for overnight batch processing. Other potential applications that



save time and processing costs, are:

- data export, in which extracts of defined sets of data are input to another DBMS, SAS, or transmission to remote sites
- sequential processes which allow extensive batch processing of ADABAS files and prevents potential conflict with on-line users

For more information about ADASTRIP, please call our ADABAS customer support

### Inside...

COBOL/370 and LE/370 Articles	2	Unisys Production System Upgrade	5
Unisys Platform Purchases Smart		CICS: Who's Behind the Scenes?	6
Console Software	4	New 'Vanilla' Environment	7
Unisys and IBM Command Centers to Co-Locate	5	Natural DBLOG Test Utility	11

---

# COBOL/370 and LE/370 Articles



by Gary Duffield

**T**his is the second in a series of eight articles discussing features of the COBOL/370 and LE/370 programming languages. This article discusses the use of the **INLINE PERFORM**. This is not new to COBOL/370; but it is very powerful.

The concept of ‘structured programming’ has been around for quite some time. It calls for streamlined code with just one entry and one exit point. The intent is to

put an end to the old ‘spaghetti bowl’ programming style, making code much easier to maintain, and more efficient.

The **INLINE PERFORM** of COBOL/370 supports this philosophy nicely. It also gives the language a version of the **DOWHILE** and **DUNTIL** constructs.

Consider the following typical VS/COBOL code:

```
05 EOF-SWITCH                                PIC X  VALUE 'N'.
      88 AT-END                               VALUE 'Y'.
...
OPEN INPUT FILE-IN.
PERFORM READ-INPUT-FILE.
PERFORM PROCESS-FILE UNTIL AT-END.
CLOSE FILE-IN.
STOP RUN.
READ-INPUT-FILE.
      READ FILE-IN
      AT END MOVE 'Y' TO EOF-SWITCH.
PROCESS-FILE.
      IF RECORD-TYPE = '1'
      [logic].
      IF RECORD-TYPE = '2'
      [logic].
      PERFORM READ-INPUT-FILE.
```

Small snippets like this may be easy to follow, but tracing **PERFORM** branches in a large listing can hinder comprehension of the logic flow.

*(Continued on page 3)*

---

# COBOL/370 and LE/370 Articles

(Continued from page 2)

Look at the same code using `INLINE PERFORM`:

```
OPEN INPUT FILE-IN.  
READ FILE-IN  
    AT END MOVE 'Y' TO EOF-SWITCH.  
PERFORM UNTIL AT-END  
    IF RECORD-TYPE = '1'  
        [logic]  
    END-IF  
    IF RECORD-TYPE = '2'  
        [logic]  
    END-IF  
    READ FILE-IN  
    AT END MOVE 'Y' TO EOF-SWITCH  
END-PERFORM.  
CLOSE FILE-IN.  
STOP RUN.
```

The new code eliminates the problem of following the `PERFORM` branches. Everything flows logically and results in more efficient executable code.

This is a `DOWHILE` rather than a `DUNTIL` construct, because the condition is tested *before* the logic is performed. That means that the logic may not be performed at all as in the case of an empty input file in the above example.

If you want the logic to be performed at least once, (a `DUNTIL`) you need the condition to be tested *after* the logic is performed. In that case substitute:

`PERFORM UNTIL AT-END WITH TEST AFTER`

Of course, there is still a time to use the old branch `PERFORM`. Notice that the `READ` statement is coded twice in our 'INLINE' example.

In the case of logic that is needed in many places in the program, a branch `PERFORM` still makes sense.

More information about the `INLINE PERFORM` statement can be found in:

- IBM SAA AD/Cycle COBOL/370 Programming Guide (SC26-4767)
- IBM SAA AD/Cycle COBOL/370 Language Reference (SC26-4769)

The remaining articles of this series will be published in future issues of the *DIS Technical Broadcast*.

If you have any questions about these articles, please contact Gary Duffield at 902-3031. If you would like to obtain a copy of all eight articles, contact Charie Martin at 902-3112. 📧

---

# Unisys Platform Purchases Smart Console Software

by Bill Peterson

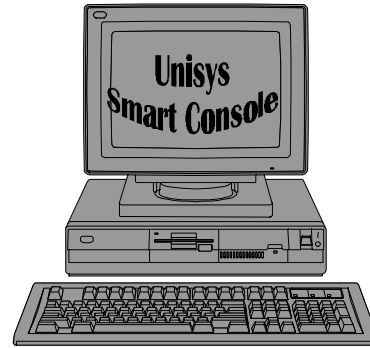
**A**utomation comes to the Unisys platform. On May 1, 1995, the Unisys Data Center purchased Smart Console, an auto-mated console software package.

Smart Console features include:

- Suppression - suppressing messages gives the console operator a cleaner view of what the system is doing by only displaying messages pertinent to their job responsibilities.
- Assigning specific - certain messages can be directed (assigned specific) to the command center console, tape, print, and workload services terminals displaying only those messages of interest to that area.
- Highlighting system messages - highlighting and flashing messages helps draw attention to important messages that require operator awareness and intervention. This is useful for supporting smoother operation of the DMRs (database maintenance backup files), preventing transaction processing queue build-ups, and system slow-downs. Smart Console also provides the ability to monitor system performance and automatically notifies designated areas when thresholds are met and action must be taken.

Current uses of Smart Console are:

- Monitoring low transaction rates and transaction queue file buildups, and simultaneously notifying the command center analyst, the Facilities Management area, and the Help Desk of a potential system slow-down as it



occurs.

- Monitoring all customer production jobs for abnormal aborts causing a message to be sent directly to a Demand terminal in the customer's Production Control area. This eliminates any missed aborts, reduces service inconsistencies, and gives the customer immediate notification, thereby providing better customer service.

A project to interface Smart Console with UOSS (unattended operations) is underway which will allow us to boot the system four to five times faster by doing it manually with little or no analyst intervention. This will also eliminate "fat-finger" mistakes and provide consistency and reliability.

Other capabilities include analyzing DASD problems by examining the sense bytes displayed and taking further action dependent on the sense information. One possible action may be automatic dialing of the customer engineer's pager for immediate service.

We are very excited about Smart

---

# Unisys and IBM Command Centers to Co-Locate

Peterson

by Bill

**A** project is underway to co-locate both Command Centers in what is currently the COM (computer output microfiche) room. The COM equipment will be relocated to the East computer room after the disposal of the DASD equipment currently in that area.

We plan to move the IBM terminal enclosure to the new area and have an addition built onto it that will house the Unisys equipment and give the new combined Command Center a horseshoe effect with 45 degree corners. The room will have the air handlers removed and replaced by a quiet unit installed above the

ceiling. The floor will be carpeted and windows placed in the doors. The combined command center is scheduled to be operational by the end of August 1995.

Although there are no immediate plans for cross-training of IBM and Unisys staff members, some sharing of knowledge will occur by simply being in the same area, looking at each other's consoles, and asking questions.

This project will encourage better communication between the data centers, provide better customer service, and give the Computer Services Division a

---

## Unisys Production System Upgrade

 by Bill Peterson

**O**n January 30, 1995, an RFQ was issued for a processor upgrade for the 2200/622ES Unisys production system to a 2200/633ES. The current system was running at nearly 100 percent capacity with transaction workload queuing and transaction response times becoming longer. The upgrade was to include an instruction processor and a channel processor that are tightly coupled and not sold separately.

Vector Technology was the successful bidder and delivered the equipment on March 24, for installation on April 2, 1995, during our normally scheduled preventive

maintenance time slot. The installation was transparent to our customers except for a noted increase in throughput. Batch processing is running during the day shift with no degradation to the system's performance.

This single processor upgrade extends the life of the production system by approximately three years. At that time we will reassess our needs in light of LAMP, ACES, and the hardware market.

If you have any questions about these three articles, please call Bill Peterson at

---

# CICS: Who's Behind the Scenes?

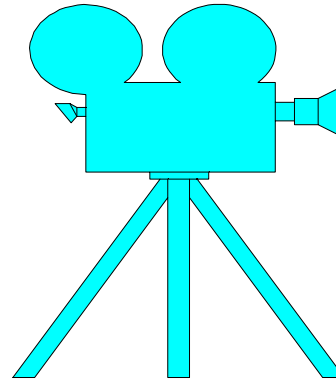


by Carol Criscione

**H**ow does a CICS problem get resolved? What happens after the DIS Network Support Center (NSC) Help Desk is contacted? A lot! But it is not always obvious to the end user what goes on “behind the scenes.” This article introduces some of the people who resolve a CICS problem.

Who is typically involved in resolving a CICS problem? The NSC Desk (753-2454) should be the first DIS telephone number a customer dials with a CICS problem. They do initial problem diagnosis by checking the network, operating system, CICS, ADABAS, TSO, etc. These folks have a *wide* variety of experience. They also have access to logs, CICS regions, and monitoring facilities like Netview and Omegamon/CICS. They resolve over 94 percent of the calls they receive without assistance, and they are striving for 100 percent! I find these people to be very dedicated to providing quality service. Recently, they have been practicing how to “kill” CICS loops to enable them to restore service to you, the customer, in the shortest time possible.

Occasionally, they will require assistance with a CICS problem. They have a variety of human resources to draw upon. They can canvas their co-workers, often finding someone else at the NSC Help Desk who has previously encountered a similar problem and can supply the needed information in a short amount of time. The DIS Network Control Center, Production



Services and Computer Operator Analysts are also helpful resources. In addition, many specialized software areas such as CICS, VTAM, Natural/ADABAS, and MVS have people on call 24-hours-a-day.

I have noticed a marked decrease in the number of “beeps” from the NSC Help Desk to CICS Technical Support over time. I believe our collaborative efforts of examining problems and finding solutions pays off!

Who works in CICS Technical Support? There are three of us who perform the day-to-day routine duties in CICS, including problem resolution.

Who are we? Who will likely answer in CICS Technical Support? Mike Reeves, is a name you may have heard before. He is often the first to answer the telephone. He has been associated with DIS and CICS Support for over 15 years. He excels in the coordination of changes, defining CICS resources, and assisting customers with a wide variety of problems. One of his major duties is to keep “things running smoothly.” I find

---

# CICS: Who's Behind the Scenes?

*(Continued from page 6)*

Gayle Huck, “does a little bit of everything”—installing system software, evaluating systems, defining resources, creating CICS regions, performance tuning, and assisting customers. Gayle certainly seems to enjoy solving a good mystery and gets to do that often in her position! She has been in the CICS field since 1987 and started out as an application programmer with another agency.

I began as a computer operator and “took the plunge” into CICS system programming/analysis in 1987. I perform duties similar to Gayle. We schedule ourselves so that at least one of us is always available—either by phone or pager. It’s a real challenge to keep track of over 35 (and counting) CICS regions! Currently, we maintain CICS Version 2.1.2, CICS Version 3.3.0 (“Vanilla CICS”), and related software products. Our next version of CICS, Version 4.1, is being included in project plans for next year.

John Howe manages the CICS and ADABAS sections. He began as an application programmer with another agency, joined the CICS section of DIS as a systems analyst/programmer in 1983, acquired ADABAS experience in 1990, and joined the management ranks in 1991. He performs as a senior technician and manager. He discourages “band-aid” fixes and is continually looking for ways to utilize resources—man and machine—more efficiently and effectively. He also works with Mike Reeves on strategic planning of the CICS environment.

These are the people directly involved in CICS problem resolution. Next issue, I will discuss how CICS problems are processed from the initial telephone call to the NSC Help Desk. Stay tuned! 📺

---

## New ‘Vanilla’ Environment

 by Gayle Huck

**A**s you all know, DIS has been in the process of migrating batch and CICS applications to the new “Vanilla” environment. The migration has been very challenging and complex because many new software products and processes have been introduced. The good news is that in the end, our customers will be able to reap

the rewards of a more secure and efficient environment.

During migration, we continue to improve our services and ensure that our software is at the latest release level in the “Vanilla” environment. One of the new releases we are preparing to install in our software

---

# New 'Vanilla' Environment

*(Continued from page 7)*

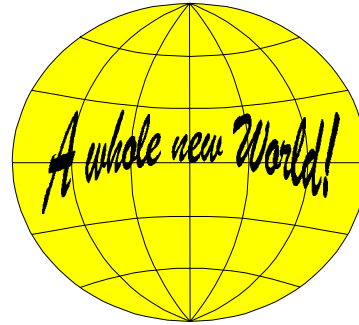
testing partition, VXRW, is CICS/ESA Release 4.1. In future articles, I will discuss the many advantages of this new release of CICS/ESA.

The Department of Social and Health Services (DSHS) was the first agency to migrate to the CICS regions running on the VIRA system. We wish to thank DSHS for their great cooperation and dedication in completing the migration process quickly and proficiently.

The Department of Labor and Industries (L&I) was the second major agency to migrate to the CICS regions running on the V4YM system. L&I accomplished a major achievement by migrating all of their CICS regions over at the same time with minor difficulties. It shows the great effort and planning L&I gave to this project.

Following L&I's migration were the Office of Financial Management (OFM) Systems (AFRS, APS, BPSI, BPSII, CAMS, TAPS, Time Accounting, A/R) and HRISD's payroll system. As of this date, these two agencies had the greatest number of COBOL programs that were converted to COBOL/370. HRISD had the added complexity of converting up to a hundred ADAMINT modules which accessed their database (ADAMINTs are no longer supported on either side).

In addition, Services for the Blind, Central Stores, State Patrol, Department of Ecology, and Department of Licensing (DOL) have also completed migration.



The Department of Natural Resources (DNR), DIS statewide applications and Department of Retirement Systems (DRS) have migrated most of their systems and are still in the migration process.

The Employment Security Department (ESD) is in the process of migrating their legacy programs to "Vanilla". ESD has many OS/VS COBOL programs and ADAMINT modules to convert. They also are currently developing major systems in the "Vanilla" environment.

The Department of Licensing (DOL) has a major new system that they have been developing on the VIRA partition. It is called the LAMP project. DOL is using CPI-C (Common Programming Interface Communications) instead of CICS/APPC (Advanced Program-to-Program Communication) to communicate from the PC workstation to the mainframe. CPI-C can be used to communicate from many different platforms at IBM and non-IBM sites. The LAMP project's database is DB2 and they will be using MVS/APPC which allows batch programs to communicate with CICS.

*(Continued on page 9)*



# New 'Vanilla' Environment

(Continued from page 8)

The "Vanilla" environment is now over the 80 percent mark in processing. We have two major partitions called V1RA and V4YM that run customer CICS regions and batch processes.

Following is a list of the CICS/ESA R330 regions running on the "Vanilla" environment:

## V1RA SYSTEM PRODUCTION/DEMO REGIONS

CICP1300 (CICP1)	DSHS/CAMIS System
CICP2540 (CICP2)	ESD
CICP3155 (CICP3)	Shared Statewide Applications -
	OFM/AFRS,
APS,	BPSI, BPSII,
	CAMS, Central Stores,
	HRISD/ PAY1, TAPS,
	Time Accounting,
	State Patrol
CICP4155 (CICP4)	Shared Production -
	DSHS/FRS, DNR,
	DRS, DOL/UCC,
	DOL/MMLS, Ecology,
	Services for the Blind
CICP5240 (CICP5)	DOL LAMP System
CICD1155 (CICD1)	Shared DEMO

## V1RA SYSTEM TEST REGIONS/QUALITY ASSURANCE REGIONS

CICX1155 (CICX1)	Test Migration -
	Unavailable to
	customers on 7/1/95
CICX2155 (CICX2)	SHARED TEST -

	AGENCY #0-250
CICX3155 (CICX3)	SHARED TEST -
	AGENCY #251-500
CICX4155 (CICX4)	SHARED TEST -
	AGENCY #501-999
CICX6540 (CICX6)	ESD legacy
	applications
CICX7240 (CICX7)	DOL LAMP
CICQA155 (CICQA)	Shared Quality
	Assurance
CICQB540 (CICQB)	ESD Quality
	Assurance
CICQC240 (CICQC)	DOL LAMP Quality
	Assurance

## V4YM SYSTEM PRODUCTION/DEMO REGIONS

CICP6235 (CICLI)	L&I MIPS
CICP7235 (CICL3)	L&I LINIIS Terminal
	Owning
Region	
CICA1235	L&I LINIIS
	Application Owning
	Region
CICA2235	L&I LINIIS
	Application Owning
	Region
CICA3235	L&I LINIIS
	Application Owning
	Region
CICD2235 (CICLD)	L&I DEMO1
CICD3235 (CICD3)	L&I DEMO2

## V4YM SYSTEM TEST REGIONS

CICX5235 (CICX5)	L&I Test
------------------	----------

(Continued on page 10)

# New 'Vanilla' Environment

(Continued from page 9)

Following is a list of the remaining CICS V212 Regions running on the Q-systems:

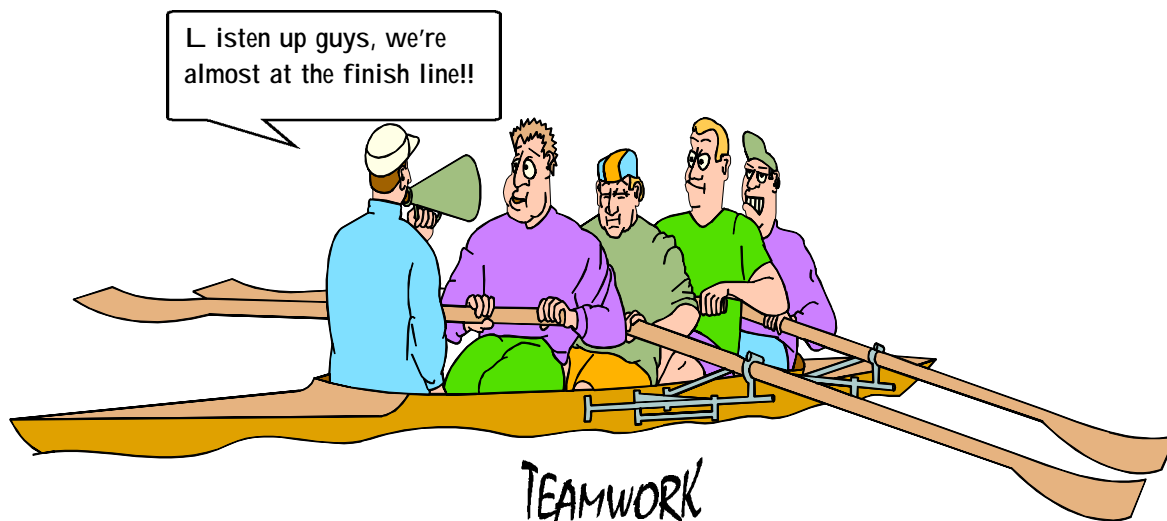
Q1CD SYSTEM PRODUCTION/DEMO REGIONS	
CICS0155 (CICS0)	Multi-User Terminal Owning Region
CICSR155 (CICSR)	DRS
CICWP155 (CICWP)	Dept of Health, Shared Natural
CICWR155 (CICWR)	DNR
CICXP155 (CICXP)	DNR
CICS1155	Shared File Owning Region
CICS3155 (CICS3)	ESD
CICDM155 (CICDM)	Shared
DEMO	
Q1CD SYSTEM	

## TEST REGIONS

TCICS155	Shared Test, ESD GUIDE test
CICET155	ESD Test
CICMT155	ESD GUIDE Quality Assurance
CICFT155	Shared File Owning Region

With the migration of DOL, HRISD, OFM, DSHS and L&I, we deleted thirteen regions from the Q-systems, two from the Q1CD system, four from the Q2EF system which has been combined with the Q1CD system due to reduced processing, and all (a total of seven) from the Q484 system.

We are almost at the end of our journey of migrating to the "Vanilla" environment. Everyone involved deserves to be congratulated for a job well done! 🎉



# Natural DBLOG Test Utility

 by Ian Heath

**T**he Natural Test Utility, DBLOG, can be found under the Development Facilities of the Natural Application Development System main menu. I have included the screens and options needed to invoke and subsequently view the log entries.

The menu selections are in **Bold print**.

```
13:53:56    ***  N A T U R A L  APPLICATION DEVELOPMENT SYSTEM  ***    95-05-25
User IH00155      - Main Menu -                               Library IH00155
                                                Mode Structured

      Code  Function
      ----  -
      D    Development Facilities
      R    Reporting Facilities
      A    Administration Facilities
      ?    Help
      .    Exit NATURAL Session
      ----  -

      Code ..  D

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
```

```
13:54:25    ***  N A T U R A L  APPLICATION DEVELOPMENT SYSTEM  ***    95-05-25
User IH00155      - Development Facilities -                     Library IH00155
                                                Mode Structured
                                                Program AUTO

      Code  Function
      ----  -
      C    Create Object
      E    Edit Object
      R    Rename Object
      D    Delete Object
      X    Execute Program
      T    DB Command Log Facility
      B    Debugging Facility
      L    List Objects, X-Ref
      G    Global Environment
      ?    Development Facilities Help
      .    Exit Development Facilities
      ----  -

      Code ..  T      Type .. _      Name .. _____

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit  C      E      R      D      X      T      L      G      Canc
```

(Continued on page 12)

# Natural DBLOG Test Utility

(Continued from page 11)

On the following menu 'B' has been selected to begin logging and 'X' was placed after the FB to capture the format buffer.

```
13:55:07          ***** NATURAL Test Utilities *****          95-05-2
User IH00155          - DBLOG Menu -          Library IH00155

      Code  Function
      ----  -
      B      Begin Logging of ADABAS Commands
      E      End and Display Log Records
      S      Snapshot of Specific ADABAS Commands
      .      Exit
      ----  -
Code ..  B

Command ..  _      Skip .....  _      Program ....  _
DBID .....  _      FNR .....  _      Line from .. 0000
Low Resp .  _      High Resp .. 9999_      Line to .... 0000

Optional Buffers for Code B
  FB ..  X RB .. _ SB .. _ VB .. _ IB .. _

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12--
      Help Print Exit Begin End Snap                                Canc
```

The following message will appear on your screen and indicates the logging is now active.

```
MORE
DBLOG started now.
```

this is only a test!

(Continued on page 13)

# Natural DBLOG Test Utility

(Continued from page 12)

Below, the selected program to be executed is AUTO.

```
13:56:22    ***  N A T U R A L  APPLICATION DEVELOPMENT SYSTEM  ***    95-05-25
User IH00155      - Development Facilities -      Library IH00155
                                           Mode Structured
                                           Program AUTO

      Code  Function
      ----  -
      C      Create Object
      E      Edit Object
      R      Rename Object
      D      Delete Object
      X      Execute Program
      T      DB Command Log Facility
      B      Debugging Facility
      L      List Objects, X-Ref
      G      Global Environment
      ?      Development Facilities Help
      .      Exit Development Facilities
      ----  -

Code ..  X      Type ..  P      Name ..  AUTO_____

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
      Help  Menu  Exit  C      E      R      D      X      T      L      G      Canc
```

To End and Display Log Records return to the DB command Log Facility and select 'E'.

```
13:57:15          ***** NATURAL Test Utilities *****          95-05-25
User IH00155      - DBLOG Menu -      Library IH00155

      Code  Function
      ----  -
      B      Begin Logging of ADABAS Commands
      E      End and Display Log Records
      S      Snapshot of Specific ADABAS Commands
      .      Exit
      ----  -

Code ..  E

Command ..  ___      Skip .....  ___      Program ....  ___
DBID .....  ___      FNR .....  ___      Line from .. 0000
Low Resp .  ___      High Resp .. 9999_      Line to .... 0000

Optional Buffers for Code B
  FB .. _  RB .. _  SB .. _  VB .. _  IB .. _

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
      Help  Print Exit  Begin End  Snap                                Canc
```

(Continued on page 14)

# Natural DBLOG Test Utility

(Continued from page 13)

For brevity, only the first three lines of the log have been displayed.

```
13:57:42          ***** NATURAL Test Utilities *****          95-05-25
User IH00155          - DBLOG Trace -          Library IH00155

M No Cmd DB  FNR Rsp      ISN      ISQ      CID  CID(Hex) OP   Pgm   Line
- 694 L2 230   2          702      ?-?? 01600101 H   AUTO   0160 Top
- 695 L2 230   2          703      ?-?? 01600101 H   AUTO   0160
- 696 L2 230   2          704      ?-?? 01600101 H   AUTO   0160
.
.
.

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Print Exit Top   Posi Bot   -      +                      Canc
```

Placing an 'F' under the column heading 'M' displays the Format Buffer.

```
13:57:42          ***** NATURAL Test Utilities *****          95-05-25
User IH00155          - DBLOG Trace -          Library IH00155

M No Cmd DB  FNR Rsp      ISN      ISQ      CID  CID(Hex) OP   Pgm   Line
F 694 L2 230   2          702      ?-?? 01600101 H   AUTO   0160 Top
- 695 L2 230   2          703      ?-?? 01600101 H   AUTO   0160
- 6
- 6          Seq No .. 694   Format Buffer
- 6 0000 * C1C46BF0 F2F06BC1 6BC1C56B F0F2F06B * AD,020,A,AE,020, * 0000
- 6 0010 * C16BC1C6 6BF0F1F0 6BC16BC1 C76BF0F0 * A,AF,010,A,AG,00 * 0010
- 7
- 7 0020 * F26BE44B 00000000 00000000 00000000 * 2,U.          * 0020
- 7 0030 * 00000000 00000000 00000000 00000000 *          * 0030
- 7 0040 * 00000000 00000000 00000000 00000000 *          * 0040
- 7
- 705 L2 230   2          713      ?-?? 01600101 H   AUTO   0160
.
.
.

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Print Exit Top   Posi Bot   -      +                      Canc
```

This is just a teaser of the DBLOG function and with a little experimentation of your own, you can determine Data Base activity in your programs.

If you have any questions about DBLOG Test Utility, please call the DIS Data Base support group at 902-3135.

# the Technical Broadcast



Charie L. Martin, Editor  
Department of Information Services  
1310 Jefferson ST SE  
Mail Stop: 42452  
Olympia, WA 98504-2452

**Technical Broadcast Staff** - The *Technical Broadcast* is published quarterly by the Department of Information Services (DIS). The purpose is to provide a forum for customer information-sharing of upcoming system enhancements and optimization tips. We invite your comments and suggestions.

If your name or address is incorrect, or if someone you know wishes to be added to or deleted from the distribution list, please fill out the form below and send via campus mail to the above address. Thank you.

Department of Information Services (DIS) is an equal opportunity employer and does not discriminate on the basis of race, religion, color, sex, age, nationality, or disability.

✂

ADD	<input type="checkbox"/>	DELETE	<input type="checkbox"/>
Name: _____			
Agency: _____			
Address: _____			
City: _____ State: _____ Zip Code: _____			
Mail Stop: _____		Phone No.: _____	

✂